



DWF_dp_absval function

Absolute Value

Version, STAR, and myDesignWare Subscriptions: [IP Directory](#)

Description

The DWF_dp_absval function returns the absolute value (magnitude) of argument a. Argument a and the return value are signed (two's complement). The signed return value can be converted to an unsigned to prevent overflow. Overflow occurs for the value $\text{DWF_dp_absval}(-2^{\text{width}-1}) = 2^{\text{width}-1}$, which cannot be represented as a signed number of width bits.

Table 1-1 Function Names

Function Name	Description
DWF_dp_absval ^a	VHDL/Verilog signed (two's complement) absolute value

a. A similar function DWF_absval exists for DesignWare Building Blocks component DW01_absval. However, function DWF_dp_absval is likely to give better QoR, especially if used inside a larger datapath context.

Table 1-2 Argument Description

Argument Name	Type	Width	Description
a	Vector	width	Input data
DWF_dp_absval	Vector	width	Return value

Table 1-3 Parameter Description (Verilog)

Parameter	Values	Description
width	≥ 1	Word length of input a and return value

Verilog Include File: DW_dp_absval_function.inc

Functional Description

```
z[width-1:0] = DWF_dp_absval (a[width-1:0])
z[width-1:0] = a[width-1:0] if a[width-1:0] >= 0
               = -a[width-1:0] else
```

For more about DesignWare datapath functions, refer to [Arithmetic – Datapath Functions Overview](#).

Related Topics

- [Arithmetic – Datapath Functions Overview](#)
- [DesignWare Building Block IP User Guide](#)

VHDL Example

```
library IEEE, DWARE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;
use DWARE.DW_dp_functions.all;
-- DWARE.DW_dp_functions_arith package if IEEE.std_logic_arith is used

entity DWF_dp_absval_test is
  port (a, b, c : in  signed(7 downto 0);
        z       : out signed(15 downto 0));
end DWF_dp_absval_test;

architecture rtl of DWF_dp_absval_test is
begin
  z <= DWF_dp_absval (a * b) + c;
end rtl;
```

Verilog Example

```
module DWF_dp_absval_test (a, b, c, z);

  input  signed  [7:0] a, b, c;
  output signed [15:0] z;

  // Passes the parameter to the function
  parameter width = 16;

  // add "$SYNOPSISYS/dw/sim_ver" to the search path for simulation
  `include "DW_dp_absval_function.inc"

  assign z = DWF_dp_absval (a * b) + c;

endmodule
```

Copyright Notice and Proprietary Information

© 2022 Synopsys, Inc. All rights reserved. This Synopsys software and all associated documentation are proprietary to Synopsys, Inc. and may only be used pursuant to the terms and conditions of a written license agreement with Synopsys, Inc. All other use, reproduction, modification, or distribution of the Synopsys software or the associated documentation is strictly prohibited.

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at <https://www.synopsys.com/company/legal/trademarks-brands.html>.

All other product or company names may be trademarks of their respective owners.

Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
www.synopsys.com

